* 1. 

Readme

Windows Azure PowerShell Cmdlets

Version: 2.2.1

Last updated: 1/24/2012



## Contents

[Overview 2](#_Toc312023828)

[Getting Started 2](#_Toc312023829)

[Installing the Windows Azure PowerShell Cmdlets 2](#_Toc312023830)

[Listing Installed Cmdlets 6](#_Toc312023831)

[Displaying Cmdlet Help 8](#_Toc312023832)

[Using the Cmdlets 9](#_Toc312023833)

[Using the Windows Azure Hosted Services Cmdlets 17](#_Toc312023834)

[Changes in version 2.2 19](#_Toc312023835)

[Changes in version 2.1 20](#_Toc312023836)

[Changes in version 2.0 20](#_Toc312023837)

Overview

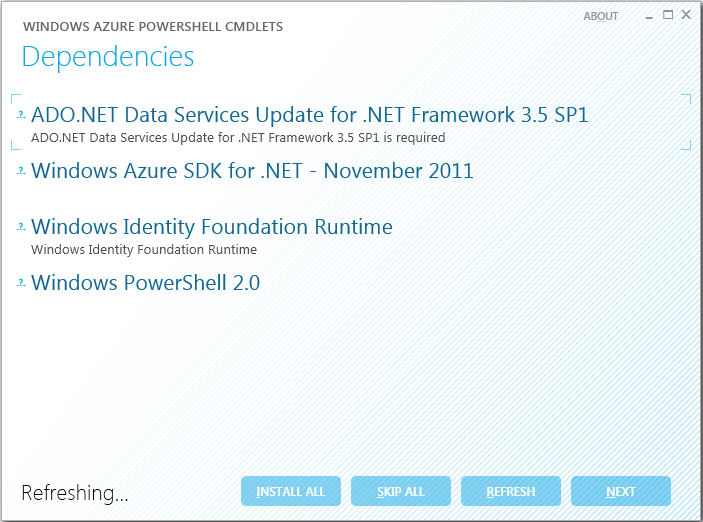
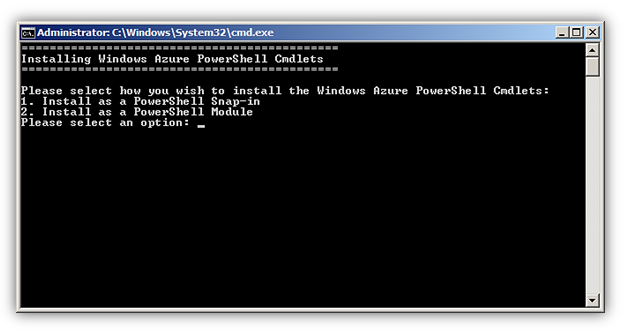
The Windows Azure PowerShell Cmdlets allow you to use a scripting interface to manage Windows Azure, SQL Azure and Windows Azure Access Control Service operations. They are designed to be usable tools for browsing, configuring, and managing several Windows Azure services.

These tools can be helpful when developing and testing applications that use Windows Azure Services. For instance, using these tools you can easily create a new deploy of your services, change configuration for a specified role, manage SQL Azure servers, etc.

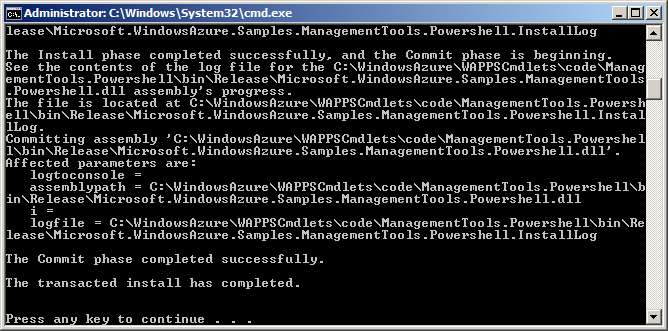
Getting Started

Installing the Windows Azure PowerShell Cmdlets

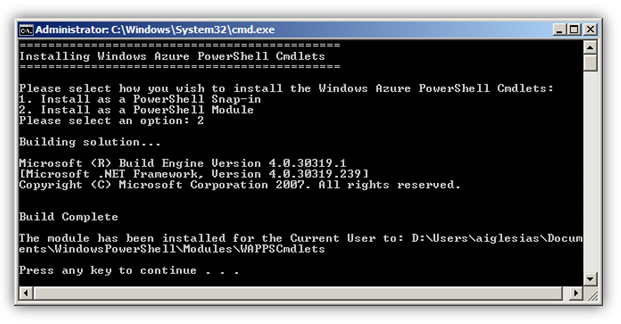
To get started using the Windows Azure PowerShell Cmdlets, complete the following steps:

* 1. Run the **StartHere.cmd** command script located in the directory where you extracted the package.
  2. The **StartHere.cmd** script will launch the Dependency Checker tool. The Dependency Checker verifies your machine to ensure that it is properly configured with all of the necessary dependencies to use the Windows Azure Services PowerShell Cmdlets.
  3. The first step involves checking your machine for the software and configuration. If you do not have the required configuration or dependencies then, in most cases, you will be provided with a link to install them using the Web Platform Installer.
     1. 
     2. Figure 1
     3. Checking for dependencies
  4. After the dependency check is complete, press the **Next** button to proceed with the installation. A console window is shown to select the installation method. Enter “1” to install the cmdlets as a PowerShell snap-in or “2” to install them as a PowerShell module.
     1. **Note:** To uninstall the Windows Azure PowerShell Cmdlets use the *uninstallPSCmdlets.cmd* script located in the *setup* folder.
     2. 
     3. Figure 2
     4. Choosing an installation mode
  5. The console window shows results differently depending on the choice you make in the previous step.

If you choose to install the Windows Azure PowerShell Cmdlets as a PowerShell snap-in, the output in the console window should be as follows.

* + 1. 
    2. Figure 3
    3. Installing the cmdlets as a PowerShell snap-in

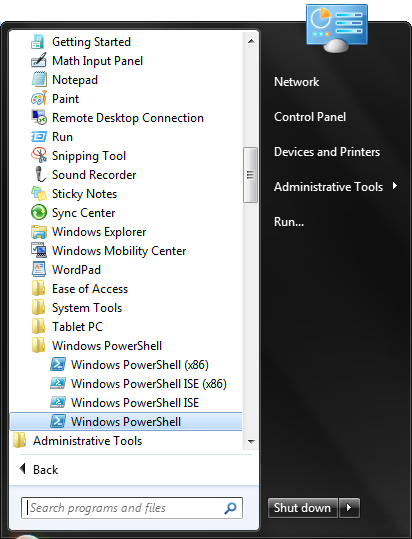
When installing the Windows Azure PowerShell Cmdlets as a PowerShell module, the output should appear as shown below.

* + 1. 
    2. Figure 4
    3. Installing the cmdlets as a PowerShell module
  1. Wait until the cmdlets project installed and then press any key to close the console window.

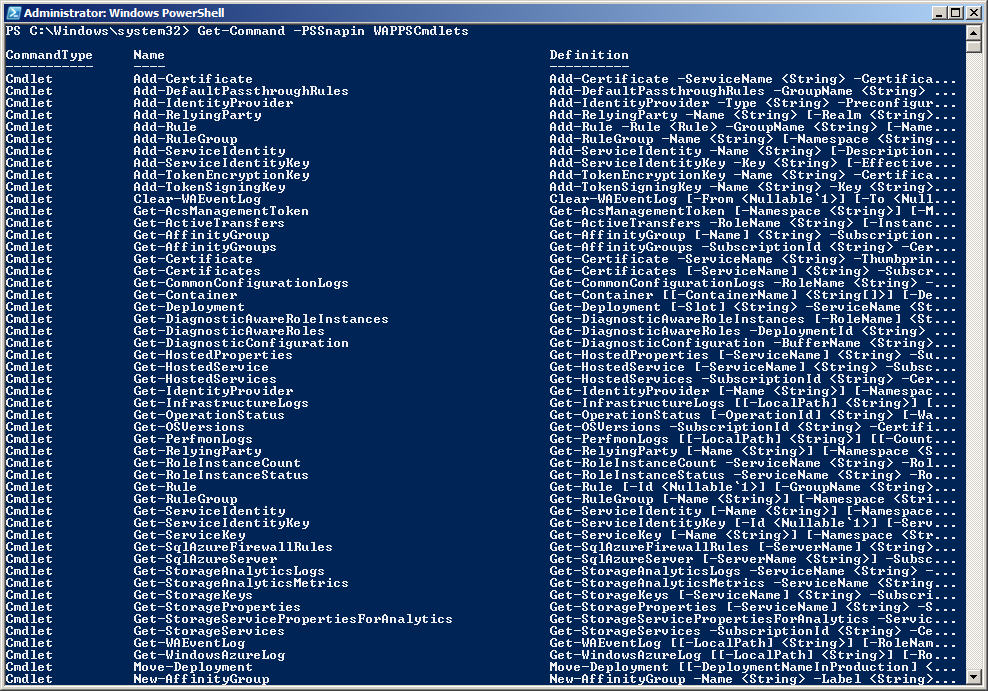
Listing Installed Cmdlets

The Windows Azure PowerShell Cmdlets perform most operations related to the administration and management of Windows Azure services and storage.

You can use a PowerShell script to list the set of PowerShell cmdlets available:

* 1. Open a PowerShell console from **Start | All Programs | Accessories | Windows PowerShell | Windows PowerShell**.
     1. 
     2. Figure 5
     3. Opening the PowerShell console
  2. Add the **WAPPSCmdlets** snap-in to your session by typing the following command:
     1. PowerShell
     2. Add-PSSnapin WAPPSCmdlets

or load the corresponding module by typing the following command:

* + 1. PowerShell
    2. Import-Module WAPPSCmdlets
  1. To retrieve a list of all the available cmdlets in the package, type the following command:
     1. PowerShell
     2. Get-Command -Module WAPPSCmdlets
     3. 
     4. Figure 6
     5. Listing available cmdlets

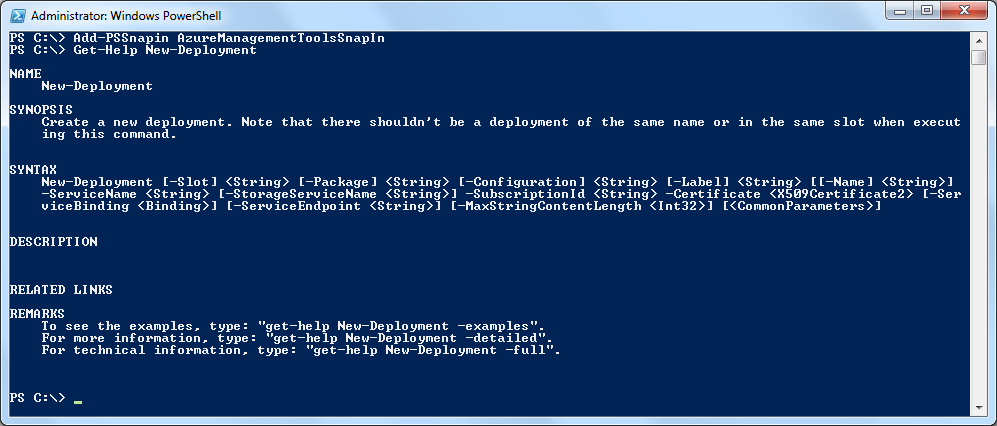
Displaying Cmdlet Help

When using individual cmdlets, you will probably need to know detailed information about how to use each cmdlet.

To get help information about a particular cmdlet, follow these steps:

* 1. Open a PowerShell console window from **Start | All Programs | Accessories | Windows PowerShell | Windows PowerShell**.
  2. Add the **WAPPSCmdlets** snap-in by typing the following command:
     1. PowerShell
     2. Add-PSSnapin WAPPSCmdlets

or load the corresponding module by typing the following command:

* + 1. PowerShell
    2. Import-Module WAPPSCmdlets
  1. To retrieve help information about a specific cmdlet, use the **Get-Help** command. For example, to see the help topic for the **New-Deployment** cmdlet, type:
     1. PowerShell
     2. Get-Help New-Deployment
     3. 
     4. Figure 7
     5. Retrieving help information for the New-Deployment cmdlet
  2. **Note:** To get detailed information about a particular cmdlet you can specify the **–Detailed** or **–Full** options in **Get-Help**. This will give additional information about parameters, usage examples, etc.
  3. For example:
  4. Get-Help Move-Deployment –Detailed

Using the Cmdlets

The Windows Azure PowerShell cmdlets can be used to run unattended scripts to configure and manage Windows Azure services. The cmdlets provided in this package include the following:

* 1. **Note:** Cmdlets introduced in this release are identified with the label **(new)**.

## Windows Azure

* 1. **Subscription Data**

|  |  |
| --- | --- |
| Name | Description |
| Set-Subscription **(new)** | Configures common subscription settings for the PowerShell session such as subscription ID, management certificate, and storage account credentials.  Multiple named subscriptions are supported. |
| Get-Subscription **(new)** | Lists settings for one or more subscriptions. |
| Select-Subscription **(new)** | Selects the active subscription. The configuration settings for the active subscription are used unless overridden by command line parameters. |
| Import-Subscription **(new)** | Imports the configuration settings from a .publishsettings file downloaded from the Windows Azure Management Portal. |

* 1. **Hosted Services**

|  |  |
| --- | --- |
| Name | Description |
| Get-HostedProperties | Lists the properties for the specified hosted account. |
| New-HostedService | Creates a new hosted service. |
| Get-HostedService | Retrieves a specified hosted service or all hosted services in the specified subscription. |
| Get-HostedServices | Lists all hosted services in the specified subscription. This cmdlet is deprecated. Use Get-HostedService instead. |
| Set-HostedService | Sets the label and description of the specified hosted service. |
| Remove-HostedService | Removes the specified hosted service. |

* 1. **Deployments**

|  |  |
| --- | --- |
| Name | Description |
| Get-Deployment | Retrieves details for a specified deployment. |
| Move-Deployment | Swaps the deployments in the Production and Staging slots. |
| New-Deployment | Creates a new deployment. Note that there should not be a deployment of the same name or in the same slot when executing this command. |
| Remove-Deployment | Deletes the specified deployment. Note that the deployment should be in suspended state. |
| Update-Deployment | Initiates an in-place upgrade of the specified deployment. |
| Set-DeploymentConfiguration | Changes the deployment's configuration. |
| Set-DeploymentStatus | Changes the deployment status to *Running* or *Suspended*. |
| Set-WalkUpgradeDomain | Walks the specified upgrade domain. |

* 1. **Role Instances**

|  |  |
| --- | --- |
| Name | Description |
| Set-RoleInstanceCount **(new)** | Sets the instance count for the specified role. |
| Get-RoleInstanceCount | Returns the count of instances for the specified role. |
| Get-RoleInstanceStatus | Returns the status of the specified role instance. |
| Reset-RoleInstance | Reboots or reimages the specified role instance. |

* 1. **Affinity Groups**

|  |  |
| --- | --- |
| Name | Description |
| Get-AffinityGroup | Retrieves the properties for a single affinity group or all the affinity groups in the subscription. |
| Get-AffinityGroups | Lists all affinity groups in the subscription. This cmdlet is deprecated. Use Get-AffinityGroup instead. |
| New-AffinityGroup | Creates a new affinity group in a specified data center location. |
| Remove-AffinityGroup | Deletes an affinity group in the specified subscription. |
| Set-AffinityGroup | Updates the label and/or the description of an affinity group. |

* 1. **Service Certificates**

|  |  |
| --- | --- |
| Name | Description |
| Get-Certificates | Lists the certificates for the specified hosted service. This cmdlet is deprecated. Use Get-Certificate instead. |
| Get-Certificate | Retrieves a specified service certificate or all the certificates for the specified hosted service. |
| Add-Certificate | Uploads a service certificate. |
| Remove-Certificate | Deletes the specified service certificate. |

* 1. **Storage Services**

|  |  |
| --- | --- |
| Name | Description |
| Get-StorageKeys | Displays the primary and secondary keys for the storage account. |
| Get-StorageProperties | Lists the properties of the specified storage account. |
| New-StorageKey | Regenerates the primary or secondary storage key for the specified storage account. |
| Get-StorageAccount | Lists all storage services in the specified subscription. |
| New-StorageAccount | Creates a new storage account in a subscription. |
| Remove-StorageAccount | Deletes the specified storage account from a subscription. |
| Set-StorageAccount | Updates the label and/or the description for a storage account. |

* 1. **Operating Systems**

|  |  |
| --- | --- |
| Name | Description |
| Get-OSVersions | Lists the versions of the guest operating system that are currently available in Windows Azure. |

## Windows Azure Traffic Manager

* 1. **Profiles**

|  |  |
| --- | --- |
| Name | Description |
| New-TrafficManagerProfile **(new)** | Creates a new profile for a domain name owned by the specified subscription. |
| Get-TrafficManagerProfile **(new)** | Returns profile details optionally including all its definition versions. |
| Set-TrafficManagerProfile **(new)** | Updates an existing profile, allowing it to be enabled or disabled, and optionally specifying which definition version to enable. |
| Remove-TrafficManagerProfile **(new)** | Deletes a profile and all its definitions. |

* 1. **Definitions**

|  |  |
| --- | --- |
| Name | Description |
| New-TrafficManagerDefinition **(new)** | Creates a new definition for a given profile. |
| Get-TrafficManagerDefinition **(new)** | Returns a profile definition. |

**Endpoints**

|  |  |
| --- | --- |
| Name | Description |
| New-TrafficManagerEndpoint **(new)** | Creates a new endpoint object and returns it. |
| Add-TrafficManagerEndpoint **(new)** | Adds a new endpoint to the active definition in an existing profile. |
| Set-TrafficManagerEndpoint **(new)** | Enables or disables an endpoint in a profile definition. |
| Remove-TrafficManagerEndpoint **(new)** | Removes an endpoint from the active definition in an existing profile. |

* 1. **Monitors**

|  |  |
| --- | --- |
| Name | Description |
| New-TrafficManagerMonitor **(new)** | Creates a new monitor object and returns it. |

## Windows Azure Storage

* 1. **Blob Storage**

|  |  |
| --- | --- |
| Name | Description |
| Add-Blob **(new)** | Uploads a blob to the specified container in Windows Azure blob storage. |
| Clear-Container | Removes blobs from one or more containers in Windows Azure blob storage. |
| Get-Container | Gets the blob containers in the specified storage account. |
| Save-Container | Downloads blobs from one or more containers in Windows Azure blob storage and saves them to local disk. |

* 1. **Window Azure Storage Analytics**

|  |  |
| --- | --- |
| Name | Description |
| Get-StorageAnalyticsLogs | Downloads the analytics logs for the specified service. |
| Get-StorageAnalyticsMetrics | Downloads the Windows Azure Storage Analytics metrics for the specified service. |
| Get-StorageServicePropertiesForAnalytics | Gets Windows Azure Storage Analytics properties for a storage account. |
| Set-StorageServicePropertiesForAnalytics | Sets Windows Azure Storage Analytics properties for a storage account. |

## SQL Azure

**SQL Azure Servers**

|  |  |
| --- | --- |
| Name | Description |
| Get-SqlAzureServer | Enumerates SQL Azure servers that are provisioned for a subscription. |
| New-SqlAzureServer | Adds a new SQL Azure server to a subscription. |
| Remove-SqlAzureServer | Deletes a SQL Azure server from a subscription. |
| Set-SqlAzurePassword | Sets the administrative password of a SQL Azure server. |

1. **SQL Azure Firewall Rules**

|  |  |
| --- | --- |
| Name | Description |
| Get-SqlAzureFirewallRules | Retrieves a list of all the firewall rules for a SQL Azure server. |
| New-SqlAzureFirewallRule | Updates an existing firewall rule or adds a new firewall rule for a SQL Azure server. |
| Remove-SqlAzureFirewallRule | Deletes a firewall rule from a SQL Azure server. |

## Windows Azure Diagnostics

1. **Configuration**

|  |  |
| --- | --- |
| Name | Description |
| Get-CommonConfigurationLogs | Gets the common configuration values for all logging buffers. |
| Get-DiagnosticConfiguration | Gets the configuration for the specified diagnostic buffer (DiagnosticInfrastructureLogs, Directories, Logs, PerformanceCounters, WindowsEventLogs). |
| Set-CommonConfigurationLogs | Sets the common configuration values for all logging buffers. |

**Infrastructure Logs (WADDiagnosticInfrastructureLogsTable)**

|  |  |
| --- | --- |
| Name | Description |
| Clear-InfrastructureLog | Removes Windows Azure Diagnostic Infrastructure log data from a storage account. |
| Get-InfrastructureLog | Downloads Windows Azure Diagnostic Infrastructure log data from a storage account. |
| Set-InfrastructureLog | Sets the buffer configuration for the logs generated by the underlying diagnostics infrastructure. The diagnostic infrastructure logs are useful for troubleshooting the diagnostics system itself. |

1. **Performance Logs (WADPerformanceCountersTable)**

|  |  |
| --- | --- |
| Name | Description |
| Get-PerfmonLogs | Downloads Windows Azure Diagnostics performance data from a storage account. |
| Clear-PerfmonLogs | Removes Windows Azure Diagnostics performance data from a storage account. |
| New-PerformanceCounter **(new)** | Creates an object that represents the configuration for performance counter data sources. |
| Set-PerformanceCounter | Sets the buffer configuration for performance counter data. |

**Windows Event Logs (WADWindowsEventLogsTable)**

|  |  |
| --- | --- |
| Name | Description |
| Clear-WAEventLog | Removes Windows Event log data from a storage account. |
| Get-WAEventLog | Downloads Windows Event log data from a storage account. |
| Set-WAEventLog | Sets the buffer configuration for Windows event logs. |

**Windows Azure Trace Logs (WADLogsTable)**

|  |  |
| --- | --- |
| Name | Description |
| Clear-WindowsAzureLog | Removes Windows Azure trace log data from a storage account. |
| Get-WindowsAzureLog | Downloads Windows Azure trace log data from a storage account. |
| Set-WindowsAzureLog | Sets the buffer configuration for Windows Azure trace logs. |

1. **File Based Logs (Directories)**

|  |  |
| --- | --- |
| Name | Description |
| Set-FileBasedLog | Sets the buffer configuration for file-based logs. |

**Monitoring**

|  |  |
| --- | --- |
| Name | Description |
| Get-DiagnosticAwareRoleInstances | Returns a list of IDs of active role instances that have a diagnostic monitor running. |
| Get-DiagnosticAwareRoles | Lists the set of roles which have successfully started at least one diagnostic monitor. |

**Log and Performance Data Transfers**

|  |  |
| --- | --- |
| Name | Description |
| Get-ActiveTransfers | Returns the set of active transfers, with associated transfer information. |
| Start-OnDemandTransfer | Starts an on-demand transfer of the specified diagnostics data buffer. |
| Stop-ActiveTransfer | Stops an active on-demand transfer with the specified transfer Id. |

## Access Control Service

**Identity Providers**

|  |  |
| --- | --- |
| Name | Description |
| Add-IdentityProvider | Adds an identity provider with which to authenticate into a relying party application. |
| Get-IdentityProvider | Retrieves the identity provider with the specified name. |
| Remove-IdentityProvider | Removes the specified identity provider. |

1. **Management**

|  |  |
| --- | --- |
| Name | Description |
| Get-AcsManagementToken | Retrieves an ACS management simple web token (SWT). |

1. **Relying Parties**

|  |  |
| --- | --- |
| Name | Description |
| Add-RelyingParty | Configures a relying party application. |
| Get-RelyingParty | Retrieves the relying party with the specified name. |
| Remove-RelyingParty | Removes the specified relying party. |

1. **Rules**

|  |  |
| --- | --- |
| Name | Description |
| Add-DefaultPassthroughRules | Generates rules in a rule group that pass through the claims available from an identity provider to your relying party application. |
| Add-Rule | Specifies how an input claim is transformed into an output claim that is delivered to your relying party application. |
| Add-RuleGroup | Creates a rule group. |
| Get-Rule | Retrieves the rule with the specified Id. |
| Get-RuleGroup | Retrieves the rule group with the specified name. |
| Remove-Rule | Removes the specified rule. |
| Remove-RuleGroup | Removes the specified rule group. |

1. **Service Identities**

|  |  |
| --- | --- |
| Name | Description |
| Add-ServiceIdentity | Creates a service identity. |
| Get-ServiceIdentity | Retrieves the service identity with the specified name. |
| Remove-ServiceIdentity | Removes the specified service identity. |

1. **Service Identity Keys**

|  |  |
| --- | --- |
| Name | Description |
| Add-ServiceIdentityKey | Creates a service identity key for the specified service identity. |
| Get-ServiceIdentityKey | Retrieves the service identity key with the specified Id. |
| Remove-ServiceIdentityKey | Removes the specified service identity key. |

1. **Service Keys**

|  |  |
| --- | --- |
| Name | Description |
| Add-TokenEncryptionKey | Adds a new token encryption certificate. |
| Add-TokenSigningKey | Adds a new token-signing certificate or key. |
| Get-ServiceKey | Retrieves the service key with the specified name. |
| Remove-ServiceKey | Removes the specified service key. |

Using the Windows Azure Hosted Services Cmdlets

This section demonstrates how to use the PowerShell cmdlets to deploy a new package to a hosted service, retrieve information about the new deployment, and finally delete the deployment.

* 1. The first step is to obtain a valid X.509 certificate with a key size of at least 2048 bits that can be used for authentication with the Windows Azure Management API.

One option for issuing a self-signed certificate that can be used for this purpose is to use **makecert.exe** (which ships with the Windows SDK) and use the following command to create the certificate file.

* + 1. Command Line
    2. makecert -r -pe -a sha1 -n "CN=Windows Azure Authentication Certificate" -ss My -len 2048 -sp "Microsoft Enhanced RSA and AES Cryptographic Provider" -sy 24 testcert.cer
  1. Next, upload the certificate file, **testcert.cer** in the example above, to the Management Portal, by selecting the **Management Certificates** option under **Hosted Services, Storage Accounts & CDN**. Uploading the certificate file allows Windows Azure to authenticate any requests to the Management API from the holder of the certificate.
  2. Now, start PowerShell, if it is not already running, by selecting **Windows PowerShell** from the **Start** menu.
  3. In the Windows PowerShell console, enter the following command to add the Windows Azure PowerShell cmdlets to the current session, if you have not already done so.

To load the cmdlets as a snap-in:

* + 1. PowerShell
    2. Add-PSSnapin WAPPSCmdlets

To load the cmdlets as a module:

* + 1. PowerShell
    2. Import-Module WAPPSCmdlets
  1. Now, deploy a new package using the **New-Deployment** cmdlet. The following command creates a new deployment named *TestDeploy* in the staging slot. You can obtain the subscription ID of your account and the API certificate thumbprint from the Windows Azure Management Portal.
     1. PowerShell
     2. New-Deployment -subscriptionId %SubscriptionId% -certificate (get-item cert:\CurrentUser\MY\%thumbprintInUpperCase%) -serviceName %serviceName% -slot staging -package http://%storageServiceName%.blob.core.windows.net/%container%/testPackage.cspkg -configuration config\TestServiceConfiguration.cscfg -name TestDeploy -label TestLabelStaging -storageservicename %StorageServiceName%
     3. **Note:** If the storage account name is the same as the hosted service name, the StorageServiceName parameter can be omitted.
  2. Run the following command to retrieve the deployment created in the previous step.
     1. PowerShell
     2. Get-HostedServices -subscriptionId %SubscriptionId% -certificate (get-item cert:\CurrentUser\MY\%thumbprintInUpperCase%) | where {$\_.ServiceName -eq "%serviceName%"} | Get-Deployment staging
  3. Finally, delete the deployment created previously in the Staging slot using the **Remove-Deployment** cmdlet.
     1. PowerShell
     2. Get-HostedService "%serviceName%" -subscriptionId %SubscriptionId% -certificate (get-item cert:\CurrentUser\MY\%thumbprintInUpperCase%) | Get-Deployment staging | Remove-Deployment

## Upgrading from Previous Releases

This document describes changes made to the Windows Azure PowerShell Cmdlets that can potentially affect scripts created using earlier releases.

### Changes in version 2.2

* New Windows Azure Traffic Manager cmdlets
* Added **Add-Blob** cmdlet to upload a blob to the specified container in Windows Azure blob storage.
* Added **Set-RoleInstanceCount** cmdlet to set the instance count for a role running in a deployment.
* Removed dependency on Windows Identity Foundation SDK
* Added **Set-Subscription, Select-Subcription**, **Get-Subscription**, and **Import-Subscription** to configure common subscription parameters in the PowerShell session.
* Added **New-PerformanceCounter** cmdlet to create an object that represents the configuration for performance counter data sources.
* New parameters **–From/FromUtc** and **–To/ToUtc** in **Clear-Container** to select which blobs to delete based on their “last modified” date
* Improved handling of pipeline parameters for **XXX-AffinityGroup** cmdlets
* **Get-AffinityGroup** now returns list of affinity groups if the **–Name** parameter is omitted. The **Get-AffinityGroups** cmdlet is now deprecated.
* **Get-Certificate** now returns list of certificates if the **–Thumbprint** and **–ThumbprintAlgorithm** parameters are omitted. The **Get-Certificates** cmdlet is now deprecated.
* **Get-HostedService** now returns list of hosted services if the **–ServiceName** parameter is omitted. The **Get-HostedServices** cmdlet is now deprecated.
* **Get-PerfmonLog** has been modified from taking a file name to a directory path. The cmdlet will now download a .blg per instance.

### Changes in version 2.1

* New parameter **-UseIpAddressDetection** added to **New-SqlAzureFirewallRule** to create rule using the requester's IP address

### Changes in version 2.0

**Updated PowerShell snap-in/module name**

This release merges the **Windows Azure PowerShell Cmdlets** and the **ACS Cmdlets for the Windows Azure AppFabric Access Control Service**—previously available as separate packages—into a single module. To reflect this change, the PowerShell module / snap-in has now been renamed to **WAPPSCmdlets**.

|  |  |
| --- | --- |
| Earlier Releases | Current Version |
| *Import-Module AzureManagementToolsSnapIn* | *Import-Module WAPPSCmdlets* |
| *Import-Module AcsManagementToolsSnapIn* | *Import-Module WAPPSCmdlets* |

**Changed Cmdlet Names**

Some cmdlet names have been renamed for consistency. The previous cmdlet names are still available but are deprecated. It is recommended that all scripts be updated to use the new names.

|  |  |
| --- | --- |
| Earlier Releases | Current Version |
| *Set-Deployment* | *Update-Deployment* |
| *Set-WindowsEventLog* | *Set-WAEventLog* |
| *Get-StorageProperties* | *Get-StorageAccountProperties* |
| *Get-StorageServices* | *Get-StorageAccount* |

**Changed Parameter Names**

Some parameter names have been renamed for consistency. The previous parameter names are still available but are deprecated. It is recommended that all scripts be updated to use the new names.

|  |  |
| --- | --- |
| Earlier Releases | Current Version |
| *StorageServiceName* | StorageAccountName  (See New-Deployment, Set-Deployment / Update-Deployment) |

**Updated Namespaces in the Cmdlets Solution**

The namespaces in the cmdlets solution have undergone substantial change. While this should not typically affect any existing scripts, it should be taken into account in the rare event that classes in the cmdlets project are being instantiated directly.